Finding Plans for Rearranging Robots in θ-like Environments

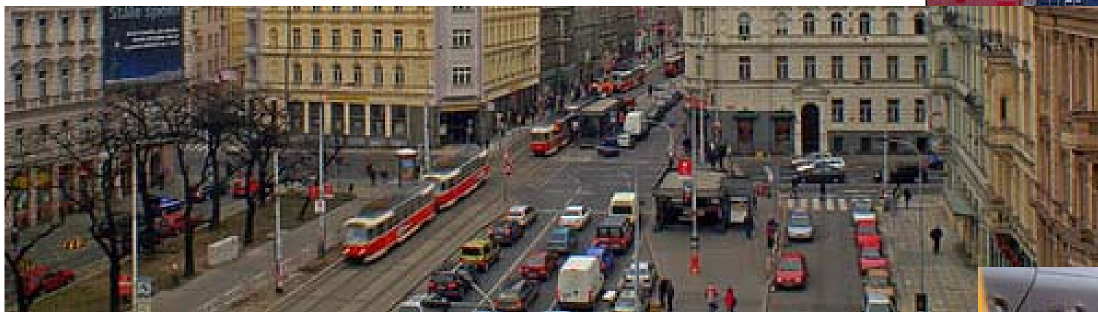
Pavel Surynek
Charles University in Prague
Faculty of Mathematics and Physics
The Czech Republic

Path Planning for Multiple Robots

- **Input:** Graph **G=(V,E)** and a set of robots $\mathbf{R=\{r_1,r_2,\ldots,r_\mu\}}$, where $\boldsymbol{\mu<|V|}$
  - **each robot** is placed **in a vertex** (at most one robot in a vertex)
  - a **robot can move into an unoccupied** vertex through an edge (no other robot is allowed to enter the vertex)
  - **initial positions** of robots ... simple function $\mathbf{S_0: R \rightarrow V}$
  - **goal positions** of robots ... simple function $\mathbf{S^+: R \rightarrow V}$
- **Task: Find a sequence of allowed moves** for robots such that all the robots reach their goal positions starting from the given initial positions

Motivation for the Problem

- Rearrangement of agents in tight space
- Automated control of heavy traffic

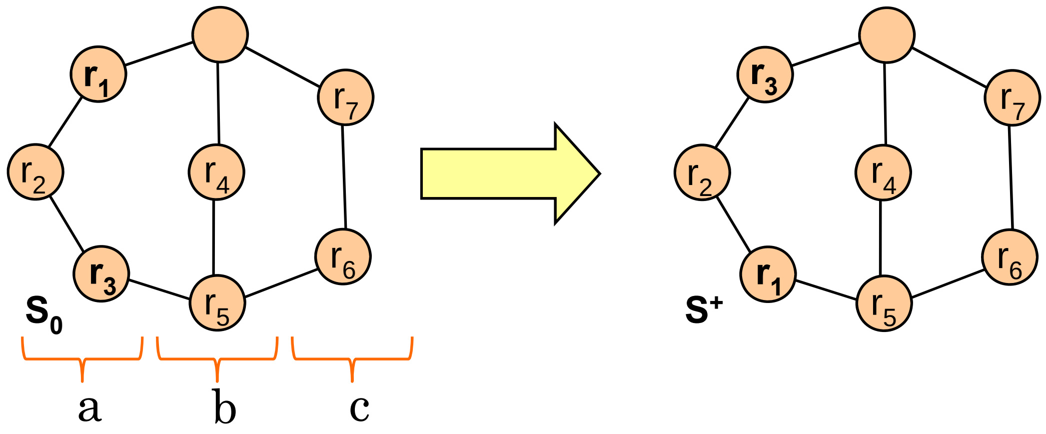


- Data transfer with limited size of the cache memory

A SPECIAL CASE WITH θ-LIKE ENVIRONMENT (θ-LIKE GRAPH)

- $G_\theta(a,b,c)=(V,E)$
- $\mu=|V|-1$
- Relation to general graphs
  - Decomposition to bi-connected components
  - Cycle decomposition of a bi-connected component
  - Last cycle with a handle represents a θ-like graph

$G_\theta(3,3,2)$



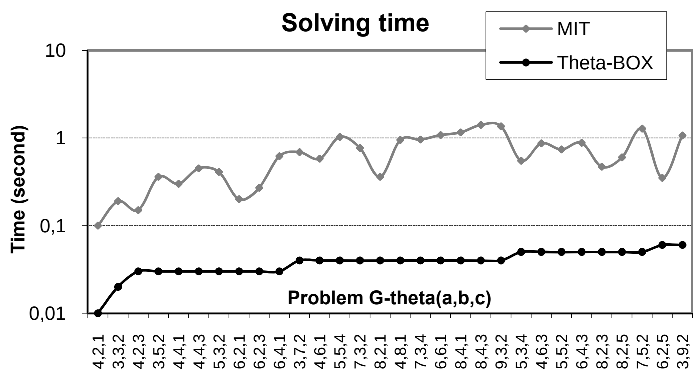
Example: Exchanging robots r_1 and r_3 (transposition of robots)

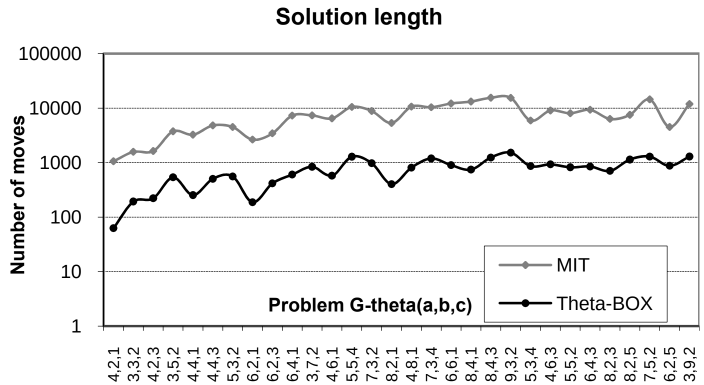
Theoretical Foundations for a Solving Method

- Interpret arrangement of robots as a **permutation**
- Proposition 1
  - **Any permutation** over μ elements can be obtained as a composition of at most μ-1 **transpositions**.
- Proposition 2
  - Any **even permutation** over μ elements can be obtained as a composition of at most μ-1 **rotations along a triple**.
- Proposition 3
  - Rotation along a **triple** is always solvable in a θ-like graph; **transposition** is solvable, if the θ-like graph contains an **odd cycle**.

A SOLVING METHOD – THETA-BOX

- **Pre-calculate** off-line optimal solutions for **transpositions** and rotations along **triples**
- **Compose a sub-optimal solution** of the optimal solutions of special cases
  - Use a fast alternative method if pre-calculated solutions are not available
  - Produces solutions of higher quality (**shorter**) than existing methods (Kornhauser et at.,1984 – MIT method)





Conclusions and Future Work

- Special case of **path planning** for multiple robots in **θ-like** environments

- **Compose a sub-optimal solution** of the optimal solutions of sub-problems (Theta-BOX method)

- Produces **shorter solutions** than existing method

- Future: Improve a process of the search for optimal solutions of the special cases (transposition, triple rotation), currently – a variant of **IDA*** is used